An Evolvable Multi-Agent Approach to Satellite Communication Systems

Sanda Mandutianu, Adrian Stoica

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91109, USA
sanda.mandutianu@jpl.nasa.gov

Abstract. This paper presents a model of a satellite communication system using evolving multi-agent architecture. Situations such as satellite communications link analysis and resource allocation, require dynamical making and breaking of links between communication nodes, and would highly benefit from increased on-board adaptability and autonomy. An evolvable architecture based on intelligent agents that communicate and cooperate with each other may offer advantages in this direction. The paper presents some problems to be addressed, a model of an evolvable system based on knowledge-based agents (software or software/hardware hybrids), and some elementary hardware building blocks that can be used to implement such a system.

1 Introduction

One of the characteristics of satellite communications is that the communication nodes are constantly moving around, and may be dynamically making and breaking links between them. The existing communication architectures use the ground stations as intermediate or even terminal nodes. An important technological problem is to configure the satellites so that they could establish the link whenever is needed and also to provide them with the capability to track one another so that they could set up the communication channel between themselves. It is not always feasible to define in advance all the possible routes, in which case the route construction has to be done in real time for each specific configuration.

We address the synergism of multi-agent model and evolutionary algorithms for a satellite communication system. The agents are communicating with each other and cooperate to achieve a common goal. While a single agent might represent a satisfactory solution for a limited complexity task, as the complexity increases, and the resources become limited, such as usually is the case on a spacecraft, the problems can be much better handled by a community of specialized agents with communicating capabilities. The multi-agent approach has been applied to various application domains such as intelligent information retrieval [1], distributed problem solving [2], etc., and has been also proposed to the spacecraft science experiment systems [3]. The evolutionary features of the agents represent the system's adaptive nature and also increase its operational robustness. Evolutionary algorithms are considered an effective way of generating or altering rules to account for changes in the physical environment. The evolutionary methods **Human** have been the object of several JPL research projects [4] [5]. The combination of evolutionary techniques and agent modeling is a novel direction in the domain of satellite communication system.

2. Proposed multi-agent architectural model

The system is comprised of a set of cooperative agents communicating with each other. Each agent receives data from the environment and exchanges data with the other agents. Each agent is a goal-oriented problem solver, possessing the abstract capability to generally approach a problem, and the necessary problem specific knowledge to solve the problem in an efficient way.

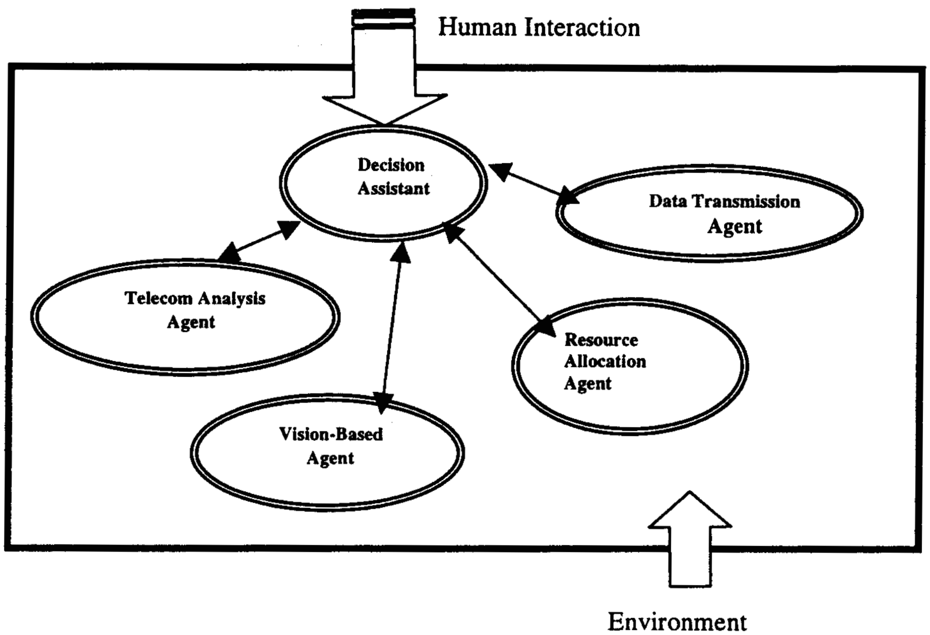

Fig.1 Multi-agent architecture

According to their functions, the agents can be defined as:

- **Decision Assistant**: supports the decision-making processes by facilitating the inter-agent communication, possesses meta-knowledge about the agent individual knowledge and their inter-relationship. Analyze the individual agent performance, and the system's performance as a whole. Makes and supports the human to make decisions at the system level: define new tasks, re-assign tasks, reconfigure the system structure.
- **Telecom Link Analysis Agent**: analyzes the telecom performance and provides telecom advisory support to the other agents;

- **Data Transmission Agent**: transforms the data to be transmitted using and adapting compression methods [6];
- **Telecom Resource Allocation Agent**: solves telecommunication resource allocation problems based on telecom constraints and data return requirements;
- **Vision-Based Monitoring Agent**: monitors changes in the spacecraft environment, informs the decision assistant about the relevant changes

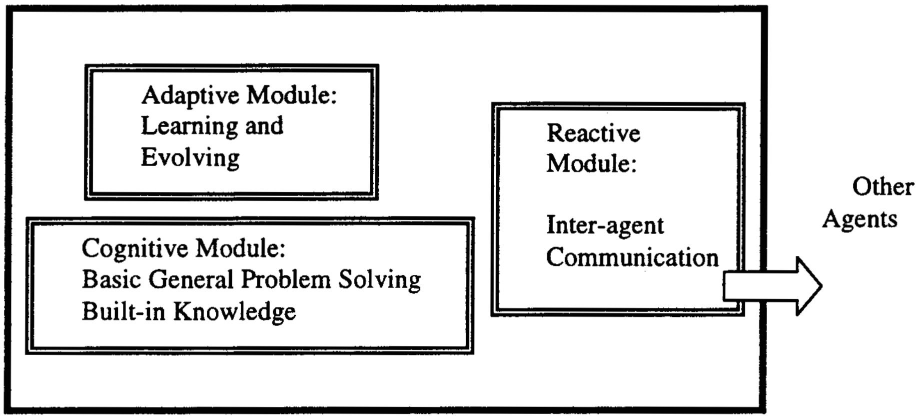

Figure 2. Agent Internal Structure

Any agent is comprised of a general problem solving mechanism and more specialized problem oriented capabilities. The general problem solving mechanism consists of a production system engine, which has the role of interpreting the rules describing the agent behavior. The rules represent the domain knowledge of the agent. Since no single set of rules is appropriate in all situations, the agent starts with a given set of rules, and evolves other, based on the interactions with the environment and with the other agents. This way, the population of problem solvers can adapt themselves to dynamically changing situations.

Due to its high potential for optimal solutions, the evolutionary techniques (such as genetic programming [7]) have been considered the appropriate method to infer and generate new rules. Upon initiation, only the basic agent functions are implemented; it is assumed that these are realized by analyzing and embedding the necessary domain knowledge before execution (mission). Problem solving requests continuously arrive at the initial agents. The agent's local decisions may affect the system's behavior as whole, and the decisions are triggered by changes in the environment. The system's behavior is analyzed by the Decision Assistant and the system might get re-organized to be more efficient. The re-organization rules can also be obtained by using evolutionary techniques: the solution obtained otherwise is not optimal, and the evolutionary approach offers a higher chance to obtain an optimal solution.

An example of reorganization might be illustrated by the case when the Data Transmission Agent is systematically overwhelmed by the requests to transmit data and is constantly behind the schedule. A solution is to first evolve another algorithm for data compression that will better perform, see for example the work on evolvable hardware for compression described in [7]. Secondly, a new agent can be defined, with the new strategy, and finally this agent will be implemented as a new hardware component of the system, for best performance.

3. Evolving agent rules

The following is an example of possible telecom agent rules. The main problem of a telecom agent is to find the best combination spacecraft-ground or spacecraft-spacecraft configuration based on telecom resource constraints and telecom or data return requirements. Another aspect is to place the necessary activities on a schedule. The antenna resources are limited and this can become a major bottleneck for establishing telecommunication links.

The following are examples of rules pertinent to this application domain:

1. The favorable tolerance for the gain using the S-band for a BWG antenna is +0.1 dB.
2. The S-band up-link is limited to 5kW transmitter power in the frequency range 2070-2090.
3. If the antenna gain is within the gain tolerance then the antenna can be used for transmission.
4. If there is more than one antenna that can be used for a link, use the one with the best data rate.
5. If there is a conflict of using the antenna at a given time, re-schedule its use.

Creating all the rules from the start is not trivial in the first place, and maintaining the rule consistency in the case of changes is another non-trivial issue. Thus, an automated method to generate or learn new rules is considered as a necessary feature. Using GP, the rules can be evolved as computer programs, or other evolutionary mechanisms could be used to search for the optimal solution.

The rules expressed in a programming language might look as in the following:

Rule: If x is X_1 and y is Y_1 then z is Z_1 (1)

For example,

```
Rule_1:
if      antenna_type= a
then
        gain_favorable_tolerance.a = x,
        gain_adverse_tolerance.a= y
        noise_temperature.a= z

Rule_2:
if  antenna_gain.a >= gain_favorable_tolerance.a and <= gain_adverse_tolerance.a
then
```

add a to feasible_antennas

The system would start using some pre-defined values (randomly generated, or, when available pre-shaped from experience) for gain_favorable_tolerance and for noise_temperature for a given type of antenna. While the results are satisfactory, there is no need to use other rules. If the telecom link doesn't perform well in time, some other values may be better.

Evolutionary algorithms operating in rule-based spaces were explored in the fields of classifier systems and evolutionary fuzzy systems. The following treatment is more from the perspective of evolutionary fuzzy systems. Several papers have demonstrated the use of evolutionary algorithms to knowledge extraction in fuzzy systems. The most common situation is when the rules are of the form (1) where X, Y, Z are fuzzy sets (most of the time the systems however implement Takagi-Sugeno reasoning and Z is not a fuzzy set but a real valued number).

In most cases the fuzzy sets are trapezoidal in which case four parameters characterize the set, or bell-shaped, in which case the parameters are the position of the peak and a measure of its spread. The evolutionary algorithm optimizes the parameters that characterize the membership function, the combination of the terms that compose the rules or the mechanisms of reasoning, choosing the operators to implement conjunctions, disjunctions, etc. In more complex cases one could choose different interpretations and reasoning mechanisms for different regions in the input space.

4. Hardware implementation of rule-based agents

Fuzzy rule-based system evaluations could be lengthy and hardware implementations ranging from RISC fuzzy processors TIL chip to analog ASIC were developed. In the following we describe a low power (~ 0.5W) high speed (1 million decisions per second) analog programmable chip designed for implementing fuzzy rule-base systems.

The fuzzy rule-base processor chip is a (digitally) programmable analog chip which implements heuristics in the form of fuzzy rules. More precisely, it implements reconfigurable sequences of MIN and MAX operations, which in the context of fuzzy logic are operators for conjunction (AND) and disjunction (OR) of variables that enter the rules. The chip is designed to process 16 input variables of 5 classes, which are combined in selected 64 rules and provide 32 decision outputs. The chip was fully simulated and a TinyChip implementing a scaled-down version was fabricated through MOSIS. The time from the moment when the inputs are presented until a decision is considered valid at the output is of the order of 1 microsecond.

The chip will work in conjunction with a Fuzzy Set Processor which will convert inputs to degrees of membership to fuzzy sets. The chip is also a digitally programmable analog chip (one can program the four parameters for the adopted trapezoidal membership functions).

We plan to use the chips to evolve fuzzy rule based systems in intrinsic EHW mode. Ultimately, such chips could be part of hybrid SW/HW evolvable agents that can control the spacecraft communication system.

5 Summary

The paper presented an evolvable multi-agent approach to sattellite communication systems. The agents are hybrid (SW/HW) rule-based. Initial work towards hardware implementation of rule-based agents and plans for on-chip evolution are described.

Acknowledgements

The research described in this paper was performed at the Jet Propulsion Laboratory, California Institute of Technology under a contract with the National Aeronautics and Space Administration.

References

1. Odubiyi, J.B., Kocur, D., J., Weinstein, S., M., SAIRE – A Scalable Agent-Based Information Retrieval Engine, Proc. of the2nd Int. Conf. on Autonomous Agents, Marina del Ray, 1997
2. Liu, J., Sycara, K.. Emergent Constraint Satisfaction through Multi-Agent Coordinated Interaction, Proc. of the 5th European Workshop on Modeling Autonomous Agents in a Multi-Agent World, Neuchatel, Switzerland, Aug. 1993.
3. Mandutianu S., A Multi-Agent System for Spacecraft Science Experiments, JPL internal document, 1997
4. Stoica, A. Fukunaga, A, Hayworth, K, Salazar-Lazaro, C. Evolvable Hardware for Space Applications, submitted to ICES98
5. Fukunaga, A, Hayworth, K, Stoica, A. Evolvable Hardware for Spacecraft Autonomy, to Appear in Proc. of IEEE Aerospace Conference, Aspen , March 21-27, 1998
6. Cheung, K. M., Tong K. : Preliminary results on intelligent telecommunication operation Strategies, Proposal for the New Millenium Technology for Communication Systems, JPL internal document, Aug.1997
7. Koza, J. R. : Genetic Programming: on the Programming of Computers by Means of Natural Selection MA:MIT Press Cambridge 1992
8. Higuchi, T. et al.: Evolvable hardware and its applications to pattern recognition and fault-tolerant systems, in Towards Evolvable Hardware Sanchez & Tomassini eds., Springer-Verlag Berlin Heidelberg 1996 p118-135